分布式事件系统研究概览

虎嵩林

摘要： 本文简单综述了分布式事件系统的概念、发展过程与主要研究内容，并介绍我所在该方面所进行的研究和应用工作。

关键字： 分布式事件系统；发布订阅；基于内容路由；复杂事件处理

1 引言

事件广泛存在于人类的活动之中，其粒度或大或小、类型包罗万象。事件既可以是生活中发生的一个现象、一条新闻，金融交易中的一个实时报价，也可以是生产环节中的一个订单，计算机中一个鼠标移动触发的消息。无论是社会、日常生活，还是计算机系统，都在持续地产生着形形色色的事件，而整个社会、个体也像计算机一样，都在被各类的事件所驱动，并参与到周而复始的事件处理活动之中。

分布式事件系统（Distributed Event Based System，DEBS）的目的就是将信息生产者提供的信息事件经过过滤或者加工处理，高效地分发给感兴趣的信息消费者，实现一个事件驱动的、松耦合的分布式环境。分布式事件系统被广泛应用在实时处理、系统监控、持续查询（continuous query）、业务流程管理等领域。这里，信息的生产者通常被称为“发布者”（Publisher），而信息的消费者则被称为“订阅者”（Subscriber），它们之间通过一个中介网络（broker network）联系起来。在具体的应用场景中，发布者可以是应用组件、数据库触发器、传感器设备、智能电网终端或者多媒体内容提供商等等；而订阅者则既可以是实时信息查询分析工具，也可以是应用组件、设备控制器、数据库等。订阅者只需要通过提交订阅（subscription）来描述其对信息消费的兴趣，而分布式事件系统的中介结点则对发布消息（Publication）与订阅之间在频道、主题乃至内容上的匹配关系进行计算，并根据匹配结果进行选择性路由，一步步将事件的信息分发给感兴趣的订阅者。

早期事件处理的研究者来自数据库、分布式系统、中间件、软件工程等诸多领域，而具有内容级过滤与处理能力的事件处理技术则吸引了多方的共同关注，充当了汇聚各个支流分派、形成独立共同体的催化剂。这种基于内容的分布式事件处理研究发端自欧洲，在十余年的发展中渐呈脉络，逐步发展成型。1998 年，意大利米兰理工大学（Politecnico di Milano）的卡扎尼格（Antonio Carzaniga）发表了其经典的博士论文《可扩展至广域网的事件通知服务系统架构（Architectures for an Event Notification Service Scalable to Wide-area Networks）》，系统地阐述基于内容的分布式事件通知系统的模型、算法和实现，标志着该研究的开端。2002 年拉克纳姆（David Luckham）教授《事件的力量：分布式企业系统中的复杂事件处理导论（The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems）》一书的出版，将复杂事件的处理技术纳入到了事件系统研究者的视野之中。2003 年，卡扎尼格同沃尔夫（Wolf）在 SIGCOOM 上首次提出了“基于内容的网络（Content based network）”[1]的概念，进一步将分布式事件系统的发展空间扩展到了网络领域。为了进一步推动分布式事件系统领域的研究工作，为研究者提供高水平的交流平台，自 2002 以来，在多伦多大学教授雅各布森（Hans Arno Jacobsen）的倡导下，分布式事件系统

的研究社团开始活跃在 SIGMOD、ICDCS、ICSE 这三个不同领域的顶级国际会议上，组织了 5 届分布式事件系统研讨（Workshop）。研究者们在围绕匹配和路由算法进行研究的同时，也关注诸如事件驱动的架构、服务计算等问题，并积极研讨分布式事件系统的教学体系和课程设计，开始形成了一个相对独立的科学共同体。2007 年，高德纳咨询公司（Gartner）发布了一项报告[2]，宣称事件处理将会带来突破性的创新，是"下一个大家伙"（Next Big Guy），事件的重要性开始得到了越来越多的重视。同年 9 月，高德纳举办了首届 Event Summit （事件处理峰会），吸引来自分布式事件系统学术界以及 IBM、甲骨文（Oracle）、微软、CA 等大型公司广泛参与。10 月，第一届 ACM 分布式事件系统会议在多伦多召开，汇集了该领域的著名学者和工业界的领军人物，标志了其研究社团的正式形成 [3]。在这段时期，分布式事件系统技术的应用也进入到了加速发展的阶段，在传统的信息分发[4]、流程管理/企业应用集成[5]、数据流管理[6]等等的基础上，进一步扩展到了博客过滤[7]、自组织（Ad Hoc）网络[8]、云计算 [9]、智能电网[10]等新兴领域。

2 主要研究内容

分布式事件系统面临的一个突出问题就是整个系统的"大规模"（Large Scale）特征。这里所指的大规模，主要有以下几个含义：1）并发事件的数量极其庞大；2）活动的订阅数量庞大，并且订阅可以随着订阅者兴趣的变化而自主更新；3）发布者或者订阅者群体的规模庞大；4）整个网络包含的节点数量多，在地理上也可能会分布在一个很大的区域。

在这样一个大规模网络的环境中， 要实现海量事件的按需、高效、可靠地分发，事件的匹配/过滤以及路由就成为系统的两个基本问题。作为一个可用的分布式系统所必需的负载均衡、容错、移动、安全等技术以及新型应用所带来的应用研究近年来也逐渐成为分布式事件系统研究的新热点。

2.1 事件匹配/过滤

事件匹配/过滤可以根据其过滤能力的强弱简单地划分为基于频道（Channel）的过滤、基于主题（Topic）的过滤和基于内容（Content）的过滤三种类型。

在基于频道的匹配方式中，事件的发布者和订阅者各自选择相应的频道标识（ID）， 事件的过滤可以直接通过标识的匹配来完成。 而在基于主题/类型（Subject/Type）的方式中，发布者和订阅者都使用一个字符串或者类型（Type）树中的一个路径来描述发布内容和订阅兴趣，二者之间匹配与否就可以通过字符串匹配或者类型之间的相等及包含关系来判定。

相比于前二者，基于内容的过滤完全依赖于事件的内容进行匹配，具有更为强大的描述和信息处理能力，是分布式事件系统研究的焦点。我们可以简单地把这样的事件系统看作成一个倒置的数据库：其中的查询（订阅）相对固定，而数据（发布）则是动态变化的。事件匹配或者过滤的目的，就是在持续的事件流上，不断获得满足一组订阅的事件集合。一般而言，支持内容匹配和过滤的事件模型通常包括属性/值对模型、XML 模型以及 RDF 模型等，不同的模型在算法处理上也有所不同。

- **属性/值对(Attribute/Value pair)模型**

属性/值对模型是最常用的模型，这个模型所描述的事件包括一组属性及其数值。典型系统如 Grypon [33]、 PADRES [34]等都是采用这样的模型。在这样一种模型中，一个发布 $\boldsymbol{P}$ 就是一个"属性-值"的集合。一个发布 $\boldsymbol{P}=\{(\boldsymbol{a}_1,\boldsymbol{v}_1),\cdots,(\boldsymbol{a}_i,\boldsymbol{v}_i),\cdots,(\boldsymbol{a}_n,\boldsymbol{v}_n)\}$， 由信息的生产者生成。其中 $\boldsymbol{a}_i$ 是属性，$\boldsymbol{v}_i$ 是对应的值，可以是包括数值型数据 、日期型、字节型、

逻辑型、字符串型乃至自定义对象等任意数据类型。订阅 S 包含一系列属性过滤条件（Attribute filters），由信息的消费者定义。每一个过滤条件 F_i（$0<i<n$， n 是自然数，表示过滤条件的数量）是一个断言 “属性-操作符-值”。可以定义一个订阅

$$S=\{(a_1,Op_1,v_1),\cdots,(a_i,Op_i,v_i),\cdots,(a_n,Op_n,v_n)\}$$

其中 a_i 是属性名称；Op_i 是对应的操作，可以包含操作“>”、“<” 、“ =”、 “belongto”等等，也可以包含字符串运算符“contains”（包含）、 “eq”（等于）等等。如果没有约束条件，还可以用“isPresent”运算符来表示该属性的存在。v_i 仍然是 a_i 对应的值。

如果一个发布 P 满足订阅 S 的所有过滤条件，则称 P 与 S 匹配。属性值对匹配的经典算法包括：滤波器算法（Brute Force 算法，也就是集合穷举算法）、计数器算法（Counting Algorithm）、决策树算法、二叉决策图算法以及两阶段匹配算法等等。

滤波器算法也就是采用简单的集合穷举的方式，将通知消息同所有的订阅进行匹配。如果订阅的所有条件都满足，则返回真，否则返回假。这个算法的思想简单、直接，但使用这种方法会导致断言的重复计算，影响整个系统的效率。

针对上述算法的不足，严（音译，Tak W. Yan）[11]提出了一种计数器算法。其中，每个断言指向多个过滤条件。每个过滤条件包含一个初始为 0 的计数器，在每一事件的匹配操作后清零。任意一个事件只同所有的断言进匹配，当某个断言被匹配时，相应的过滤条件的计数器加一。这样，当某个过滤条件的计数值等于其所包含的断言总数时，则返回匹配判定为真，否则为假。严还分别针对断言的属性、操作和值建立了三层索引，进一步提高了算法的执行效率。

阿奎莱拉（M. K. Aguilear）[12]提出的决策树算法以过滤条件作为叶子节点，“属性和操作”作为非叶子节点，所有的边则是“值”。判定过程就是对决策树的遍历过程，能够从根节点出发，满足节点和边条件到达的叶子节点返回判定值为真，否则为假。

计数器算法和决策树算法针对具有“与”关系的断言构成的过滤条件设计，无法处理包含“或”关系的过滤条件。为此，卡姆帕伊拉（A. Campailla）[13]提出了一种基于二叉决策图（BDDS：Binary Decision Diagrams）的匹配算法。其中包含两个终结节点 1 和 0。一个断言构成一个非终结节点，包含两条出边，分别是标志断言满足的高边（High Edge）和标志断言不满足的低边（Low Edge）。断言的一条出边指向下一个断言或者终结节点。他还提出了有序二叉决策图（Ordered Binary Decision Diagrams，OBDDs）结构，以实现多个过滤条件之间的共享。

两阶段匹配算法是一种高效的匹配优化算法[14]，算法分为两个阶段：第一阶段，找出至少有一个属性匹配发布的所有订阅子集；第二阶段，利用计数器的思想，计算这些订阅条件满足的次数。若全满足，则返回该订阅匹配结果为真。

- **XML/RDF 模型**

XML 模型用 XML 来描述发布，用 XPath/XQuery 来描述订阅。 阿提尼尔（M. Altinel）和富兰克林（M. J. Franklin）[15]针对 XML 信息分发的需求，提出了一种 XFilter 方法。该方法将 XPath 描述成有限自动机（Finite State Machine，FSM） 。当 XML 文件中的条件不断触发一个有限自动机，并使之到达接受状态（Accepting State）时，则认为该 XML 文件与相应的 XPath 匹配。 容易看出，由于所有的 XPath 都是独立的，XFilter 存在同滤波器算法类似的缺陷。针对这个问题，刁艳磊（音译，Yanlei Diao）同 XFilter 的作者一起又提出了 YFilter[16]算法，将所有的 XPath 的表达式合并构成一个非确定性有限自动机（NFA：Nondeterministic

Finite Automaton），以共享不同表达式之间的公共过滤条件，从而降低匹配运算次数。此外，李国莉（Guoli Li）[17]还提出了一种利用属性-值对方式实现 XML 描述和匹配的方法，能够有效地提高系统的匹配速度。

皮特洛维奇（M. Petrovic）以基于内容的 RSS（RDF Site Summery）分发需求为背景，用图模型来描述 RDF 形式的发布与订阅，提出了一个包括发布 Gp、订阅 Gs 以及本体 Ontoloty 的 G-TOPSS 模型[18]。这样，事件的匹配就被转化成了一个图匹配问题：若 Gs 与 Gp 的一个子图匹配，则认为 Gp 与 Gs 匹配。皮特洛维奇还提出了一个两层的哈希表结构，提高了系统的运行效率。

- **扩展的模型**

基于广播的模型　在简单的基于订阅的模型中，订阅信息需要洪泛到整个网络中，这往往会带来巨大的通信开销。考虑到信息发布者的数量通常会大大小于订阅者的数量，人们提出了基于广播的模型[19]，让数量较少的广播进行洪泛。广播（Advertisement）可以看作是发布者的“公告”，它同订阅一样，包含一系列属性过滤条件（Attribute filters），描述了某个发布者所发布信息中每个属性的取值范围。任一洪泛到网络上的广播 A 首先与订阅进行匹配，得到与该广播匹配的订阅子集，则广播 A 对应的发布 P 就只需与该子集进行匹配，从而降低匹配计算的量，提高匹配效率。

组合事件/订阅模型　组合事件(Composite Event)是具有关联关系的事件集合。引文[20]定义了事件并发、时间窗口内共现、选择关系、顺序关系等基本类型，并提出了一种基于有限自动机的建模方法。这种事件的检测和匹配则可以使用有限自动机的理论来求解。在基于广告的模型中，李国莉[21]提出的组合订阅（Composite Subscription）则将对事件的需求定义为多个原子订阅之间的布尔运算，以一个二叉树的结构表达，并通过单个订阅与事件的匹配以及二叉树的求解来实现对组合事件的订阅需求。二者相比，组合订阅仅仅描述了所期望事件之间的构成关系，在研究上侧重于路由算法的优化；组合事件更细致地区分时间窗口内共现、事件并发和顺序等不同的关系类型，具有更强的表达能力。

2.2 事件的路由（Event Routing）

从网络结构来看，分布式事件系统的网络包括无环（Acyclic overlay）和有环（Cyclic overlay）两种类型。无环网络的路由算法相对比较成熟，而有环网络的路由算法近期一直是分布式事件系统研究的一个焦点。从路由表结构来划分，分布式事件系统的拓扑结构包括两种：一是非结构化的覆盖网络（Overlay）；另一种则是结构化的对等（P2P）网络（也就是基于分布式哈希表的网络—DHT based Network）。前者采用与 Gnutella 网络[1]一致的拓扑结构，采用经典的路由算法。后者采用基于分布式哈希表[22]的路由方法，具体实现时多同集结点（Rendezvous Node）路由相结合。分布式事件系统的网络也可由以上几种结构加以组合，形成层次化结构或者混合结构，以满足不同的应用需求。

常见的路由算法包括以下五种类型：

- **基于订阅的路由**

如图 1 所示，该模型首先将所有订阅（比如图中的 S_1 和 S_2 ）洪泛到整个网络中，当新的发布（Pub）产生后，在中介网络（Broker）上与相对应的订阅（Sub）匹配，然后按照订阅的逆路径，将发布一步步转发到订阅所连接的中介，并由该中介将信息发送给订阅者。比

[1] 一种文件共享网络

如图 1 中，发布者 ***P*** 所产生的事件 $\boldsymbol{n}_1$ 满足 $\boldsymbol{S}_2$ 的条件 ***v***>2，$\boldsymbol{n}_1$ 就被转发给了 $\boldsymbol{S}_2$。(为了简单起见，我们在本文中假设所有的订阅和发布都只有一个属性 ***v***)。为了保障新产生的订阅尽可能地同网络中正在路由中的发布相匹配，现有的分布式事件系统在实现上往往令发布在所有经过的路径上的中介都进行一次新的匹配操作，以避免遗漏新的订阅。

- **基于广播的网路由**

如图 2 所示，在基于广播（Advertisement）的系统中，发布者在发布消息之前，需要首先发送广播消息 ***A***，将自己能够提供的信息送达给网络中的每一个中介节点。如果中介节点上存在同广播内容相匹配的订阅信息，则订阅者（比如图中的 $\boldsymbol{S}_2$）的订阅信息就会顺着广播传播方向的反方向，被一步一步地返回给发布者直接连接的节点，形成“订阅路由树”。由于 $\boldsymbol{S}_1$ 的取值 ***v***>3，不满足广播消息 ***A*** 的条件，$\boldsymbol{S}_1$ 就不会形成针对 ***A*** 的订阅路由树。这样，当发布者发布新的信息（比如图中的 $\boldsymbol{n}_1$）并且满足订阅约束条件时，发布消息就会沿着订阅路由树（比如图中的 $\boldsymbol{S}_2$ 的路由树）从叶子结点逐步回溯，被路由到消息订阅者连接的中介节点，并由该节点将消息发送给订阅者。

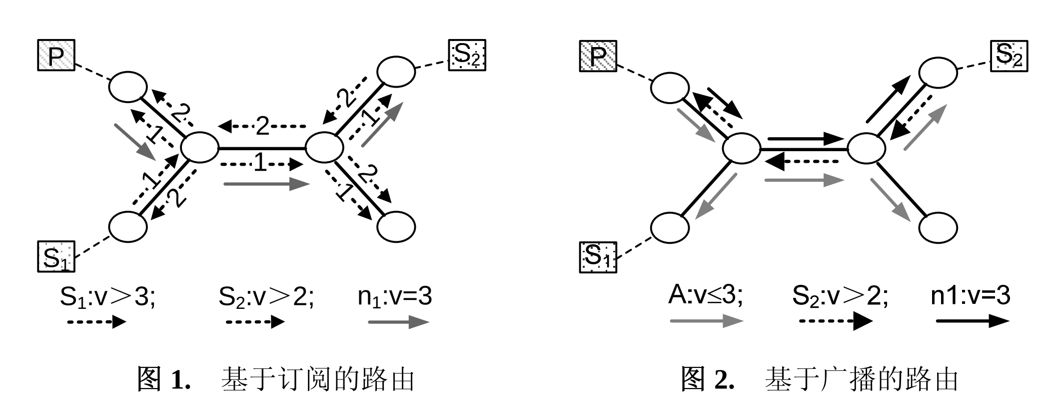


图 1.　基于订阅的路由　　**图 2.**　基于广播的路由

- **有环网路由**

卡扎尼格[23]提出了一种支持有环网拓扑结构的基于内容的路由策略。它的优势在于采用基于距离向量的方法建立订阅路径，从而使得事件能通过最短路径发送给订阅者。然而这个方案采用丢弃重复消息的方式，而不是从一开始就避免重复消息的产生，会产生大量消息副本。此外对处于环上的订阅，它们可能不会向被覆盖订阅方向转发匹配的事件（如果那个方向不是被覆盖订阅的最短路径）。李国莉[24]提出了一种新的基于内容的有环网路由协议 ECBR。它采用了带广告过滤器的方案，并为订阅的过滤器指定标识 ***id***，事件根据 ***id*** 路径的方式避免事件环路上的重复消息传递和错误。由于在 ECBR 中订阅者根据广告树建立了多条订阅路径，因此在这些订阅路径的重合点，事件能够根据当前的网络情况，采用启发式的方式找到一条最佳的路径到达订阅者，可以有效提高路由效率。

- **集结点路由**

在集合点路由结构中，存在有多个针对不同事件类型的集结点，一个集结点对应一种或者多种事件类型。符合这种类型的中介结点同该集结点一起，构成了一个独立的分发子图。整个网络还可以通过在多个集结点之间分配事件类型来实现负载均衡。

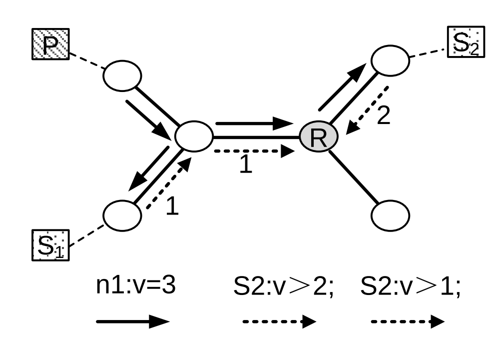


图 3.　集结点路由

在这个分发子图中，订阅也不需要进行

广播。如图 3 所示，订阅 S_1 和 S_2 首先在中介结点构成的网络中路由，直到到达一个直接连接 R 的中介结点，并被转发到 R。而发布者 P_1 发布的事件 n_1 如果在发往 R 的路径中碰到匹配的订阅，则直接沿着订阅的逆路径将 n_1 发送到订阅者。比如图中 n_1 被发送给 S_1。到达集结点 R 后，再被转发给其它的包含有满足 n_1 条件的中介结点，并最终发送给相应的订阅者。比如图中 n_1 被发送给 S_2。由于网络中的一般中介节点都需要能够找到针对特定事件类型的集结点，人们往往采用基于分布式哈希表的方法来实现集结点的发现。

- **组合事件/订阅路由**

组合事件的路由和组合订阅的路由有相似之处，引文[21]提出的组合订阅路由具有很好的代表性。当订阅者提交组合订阅后，中介根据其组合订阅中原子订阅所匹配的广播情况，将组合订阅一步步加以分解，然后分别予以路由。如图 4 所示，由原子订阅 S_1、S_2、S_3 构成的组合订阅，在中介节点 Broker 2 上先分裂出 S_3，在 Broker4 上分裂出 S_1 和 S_2。这种方法可以有效降低组合订阅的转发开销。

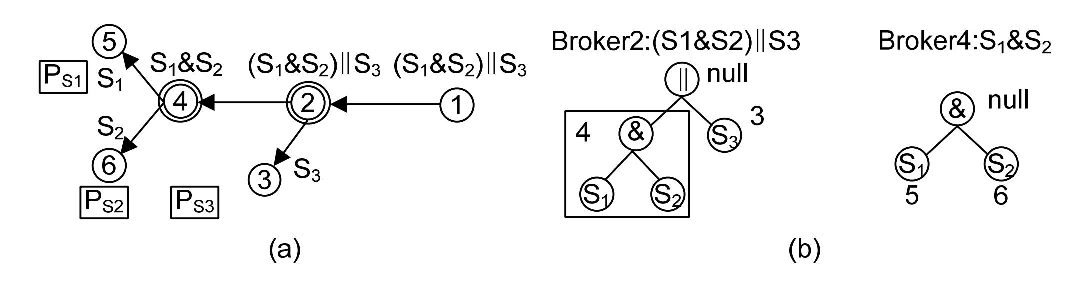


图 4. 组合订阅的分解路由方法（来源[21]）

2.2.1 路由优化

- **基于覆盖的路由（Covering based Routing）**

若订阅 S_1 的过滤条件包含 S_2，并且 S_1 的订阅路径包含 S_2 的订阅路径，则订阅 S_1 能够代表 S_2 的订阅兴趣，即订阅 S_1 覆盖（Cover）S_2。如图 5 所示，S_1 的 v 取值大于 1，S_2 的 v 取值大于 2，则 S_1 覆盖 S_2。在这种情况下，我们在图中交汇点就只需要继续转发订阅 S_1，而不需要转发 S_2。在共享路径上，对于任意匹配的消息（比如图中 $n_1(v=3)$），S_1 代替 S_2 接受消息。这样既可以有效减少订阅转发消息量，又可以压缩路由表空间，提高匹配效率。

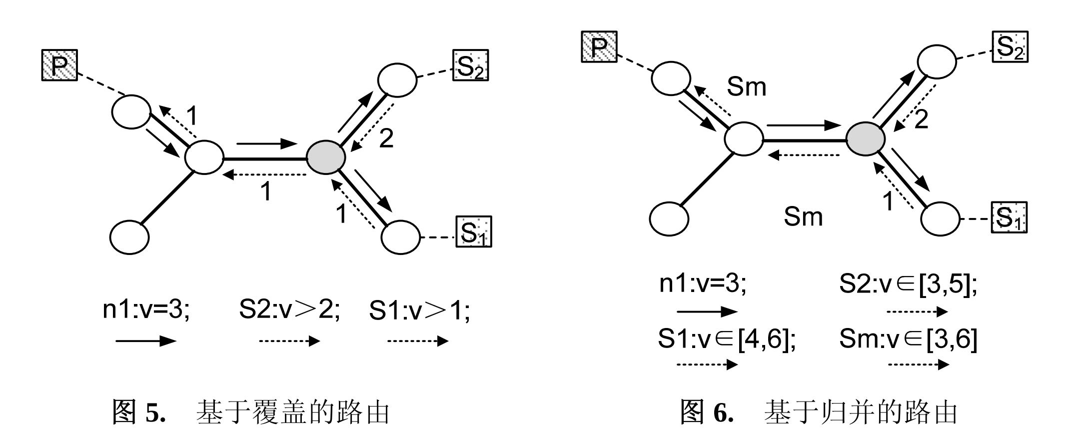


图 5. 基于覆盖的路由 **图 6.** 基于归并的路由

- **基于归并的路由（Merge Based Routing）**

基于归并的路由同基于覆盖的路由具有一定相似之处[25]。其基本思想是找出一组订阅 $\{S_i\}$ 的过滤条件的并集，生成一个新的订阅 S_m 来作为 $\{S_i\}$ 的代理，

$$\boldsymbol{S_m} = \boldsymbol{S}_1 \cup \boldsymbol{S}_2 \cdots \cup \boldsymbol{S_i} \cdots \cup \boldsymbol{S_n}$$

这样，在 $\boldsymbol{S_m}$ 与 $\{\boldsymbol{S_i}\}$ 中任意订阅的重合路径上，$\{\boldsymbol{S_i}\}$ 都可以中止转发。如图 6 所示，若 $\boldsymbol{S}_1$ 的 $\boldsymbol{v}$ 值属于[4,6]，$\boldsymbol{S}_2$ 的 $\boldsymbol{v}$ 值属于[3,5]，我们就可以生成一个并集 $\boldsymbol{S_m}$: $\boldsymbol{v}$ 属于 [3,6]来替代 $\boldsymbol{S}_1$ 和 $\boldsymbol{S}_2$ 路由。然而，在多个属性值对的复杂订阅集合上求并集开销大，且如何确定可以归并的订阅组的规模，以达到最佳的归并效果也是一个困难的问题。同时，归并路由有时会导致垃圾信息：与归并后的 $\boldsymbol{S_m}$ 匹配，但与群组中任何成员订阅都不匹配的发布事件也会被路由到 $\boldsymbol{S_m}$ 的起点，带来不必要的消息传递开销。因此，相对于基于覆盖的路由，基于归并的路由较少使用。

- **兴趣聚集（Interest Clustering）**

如果相似的订阅共享路径，则采用基于覆盖的路由优化方法可以大大减少发布消息的数量。基于这个特征，引文[26][27]研究了基于订阅者集群的方法。其基本思想是将整个系统的订阅者按照订阅的兴趣分成若干个组[27]，并将这些订阅以组为单位重新聚集，并尽可能平均地分派到临近的中介（Broker）上，从而达到提高覆盖路由优化效果的目标。

2.3 其它研究主题

在组合事件的基础上，部分研究者进一步考虑了多事件流之上更为复杂的关联关系和对更强大处理能力的需求，开始深入研究复杂事件处理技术。复杂事件处理技术从系统的角度出发加入事件聚合、分裂以及简单计算等操作算子，并结合数据库中持续化查询的部分理念，引入了“滑动窗口”等新的时间或者长度的窗口，更好地满足了实际的应用需求。XML 事件匹配算法——YFilter——的提出者刁及其团队在复杂事件处理模型、语言以及算法等方面作了一系列具有广泛影响的研究工作[28][29]，其他学者也在射频标识（RFID）应用[30]、分布式环境下的优化[31]等方面作出了贡献。分布式事件系统其它的研究主题还包括：算法的硬件加速、安全、容错处理、事务处理、负载均衡等等。

3　我们的工作

3.1 主要研究工作

围绕分布式事件处理的核心技术及其应用模型，我们开展了多方面的研究工作，在 OOPSLA、ICDCS、DEBS 等软件工程、分布式系统相关领域顶级或者重要会议上发表了一系列的学术长文。

在匹配算法方面，我们提出了基于社交和概念集成网络的事件匹配和路由方法，并将之应用在网络编程论坛的问题路由中，有效提高了问题的匹配效率。相关论文成为继 IBM 中国研究院之后，第二次被 OOPSLA 所录用的来自中国大陆的文章。在路由优化方面，我们针对有环网缺乏覆盖优化路由方法的问题，提出了有环网下的覆盖路由协议并进行了形式化的证明；发现了覆盖优化算法在取消覆盖关系时易产生巨量瞬间消息的严重缺陷，提出了一种选择性取消覆盖路由算法，显著提高了取消覆盖的算法效率，解决了系统的瓶颈问题。我们还针对分布式系统中的负载均衡和任务迁移等问题，重点研究了客户端可靠迁移协议及路由表重配置优化算法、分布式事件系统的优化部署方法等等。我们将分布式事件系统技术应用到面向服务的体系结构（Service-oriented architecture，SOA）研究中，提出基于事件的分布式资源/服务发现与组合方法，有效提高了运行效率。将部分算法思想应用到了海量服务的处理中，以 5 组 15 项全项满分的成绩赢得了 2009 年国际 web 服务竞赛第一名并在 2010 年蝉联第一。

3.2 智能电网应用

智能电网实时事件处理是分布式事件系统应用的新的生长点，也是我们研究组的应用重点。

随着智能电网建设工作的快速推进，智能电表、分布式清洁能源、电动汽车以及智能家电等得到了广泛应用。在电网末梢用电环节，巨量的新型终端设备以及设备与电网的双向互动催生了海量、实时、高频度发生的电网事件。仅我国一个省级试点项目，就涉及数千万终端、几十万中介结点，一个采集周期的事件处理规模达到亿级。为了应对这些挑战，我们同世博会智能用电展厅的承建方——国网信息通信公司——开展了技术合作，在前期研究原型的基础上，结合信通公司丰富的智能小区建设经验和迫切需求，开发了高通量事件处理平台 LIT(fLexible hIgh Throughput event Processing platform for Smart Grid)并开始进行试点应用。

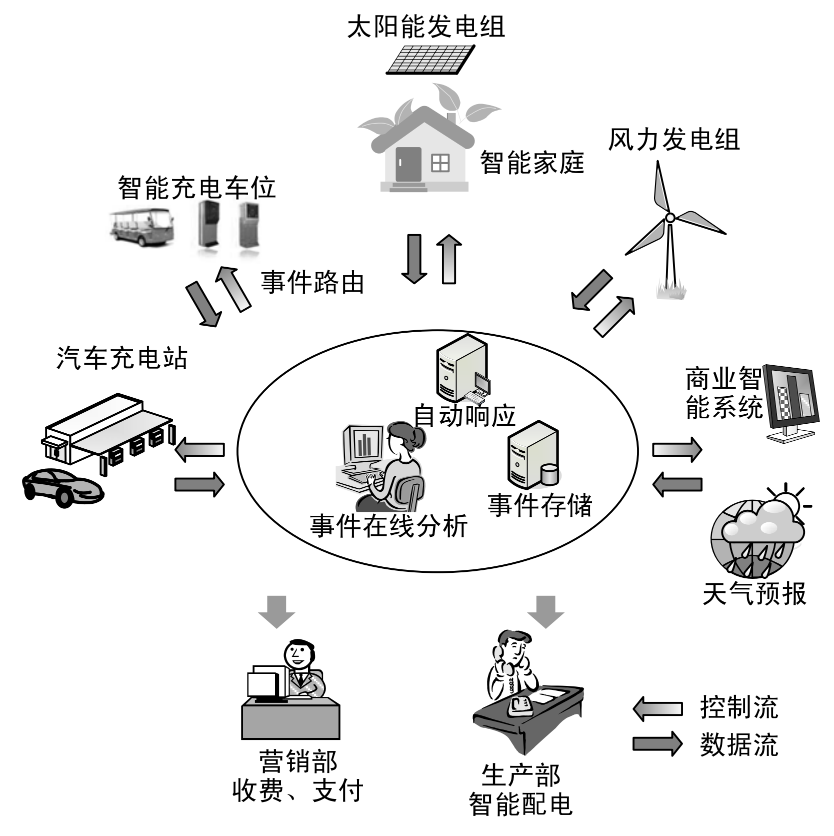


图 7. LIT 平台示意图

LIT 有点燃、点亮、高举等意思，意在切合“创新，点亮梦想”这一世博会国网馆的口号。LIT 支持海量智能电网设备端点的灵活接入，提供了一个大规模事件的过滤、高效路由和分布式存储的网络化环境，同时支持复杂事件的在线分析处理，为巨量、大并发事件在接入、传输、存储、分析、监测以及响应、控制等全生命周期所涉及的信息处理技术提供共性基础支撑。

4 总结与展望

经过若干年的发展，分布式事件系统领域已经初步发展形成了一个活跃的学术研究社团、一套初步成型的理论体系，来自不同关联领域的试验性应用也正在如火如荼地开展，形成了可喜的形势。目前的基础性的研究工作，既能为未来分布式事件系统技术的大规模应用

奠定良好的基础，也已经并将继续为其它关联学科提供许多具有参考意义的算法和协议。

尽管整个分布式事件系统技术研究已经初具规模，但其核心的关键技术和应用模式仍在不断探索之中。企业界普遍认为我们对于分布式事件系统仍然处在“缺乏充分了解（Not Fully Understood）”[32]的阶段。在这种背景下，研究分布式事件系统中的核心算法以及其新型应用模式具有明显的前瞻性研究和应用价值。

参考文献：

[1]. Carzaniga, A. and Wolf, A. L. 2003. Forwarding in a content-based network. SIGCOMM 2003. Karlsruhe, Germany: 163-174.

[2]. Massimo Pezzini, Yefim V. Natis. 2007. Trends in Platform Middleware: Disruption Is in Sight, Gartner Report.

[3]. Peter Pietzuch, Gero Muhl, Ludger Fiege. 2007. Distributed Event-Based Systems: An Emerging Community, DEBS2007. Toronto, CA: 2-

[4]. Pardo-Castellote, G., 2003. OMG Data-Distribution Service: architectural overview, In Proceedings of ICDCS Workshops, 2003. Providence, Rhode Island, USA: 200-206,

[5]. Guoli Li, Vinod Muthusamy, and Hans-Arno Jacobsen. January 2010. A distributed service-oriented architecture for business process execution. *ACM Trans. Web* 4, 1: 1-33

[6]. Vibhore Kumar, ZhongTang Cai, Brain F. Cooper, etc. 2006. IFLOW: Resource-aware Overlays for Composing and Managing Distributed Information Flows, EuroSys 2006. Leuven, Belgium

[7]. I. Rose, R. Murty, P. Pietzuch, J. Ledlie, M. Roussopoulos, and M. Welsh. 2007. Cobra: Content-based Filtering and Aggregation of Blogs and RSS Feeds. In Proceedings of the 4th USENIX Symposium on Networked Systems Design & Implementation. Berkeley, CA, USA: 29−42.

[8]. L. Mottola, G. Cugola, and G. P. Picco. Aug. 2008. A Self-Repairing Tree Topology Enabling Content-Based Routing in Mobile Ad Hoc Networks. IEEE Transactions on Mobile Computing, 7(8):946−960.

[9]. Brian F. Cooper, Raghu Ramakrishnan, Utkarsh Srivastava, Adam Silberstein, Philip Bohannon, Hans-Arno Jacobsen, Nick Puz, Daniel Weaver, and Ramana Yerneni. 2008. PNUTS: Yahoo!'s hosted data serving platform. Proc. VLDB Endow. 1, 2 (August 2008): 1277-1288

[10]. Sood, V.K, Fischer D, Eklund J.M, etc. 2009. Developing a communication infrastructure for the Smart Grid. IEEE 2009 Electrical Power & Energy Conference. Montreal, Quebec, Canada: 1-7.

[11]. Yan T.W., Garcia-Molina, H. 1994. Index structures for information filtering under the vector space model, ICDE. Houston, Texas: .337-347.

[12]. Aguilera, M. K., Strom, R. E., Sturman, D. C., Astley, M., and Chandra, T. D. 1999. Matching events in a content-based subscription system. PODC 1999. New York, NY: 53-61.

[13]. Campailla, A., Chaki, S., Clarke, E., Jha, S., and Veith, H. 2001. Efficient filtering in publish-subscribe systems using binary decision diagrams. ICSE 2001, Washington, DC: 443-452.

[14]. Fabret, F., Jacobsen, H. A., Llirbat, F., Pereira, J., Ross, K. A., and Shasha, D. 2001. Filtering algorithms and implementation for very fast publish/subscribe systems. SIGMOD 2001, New York, NY: 115-126.

[15]. Altinel, M. and Franklin, M. J. 2000. Efficient Filtering of XML Documents for Selective Dissemination of Information. VLDB2000: 53-64.

[16]. Diao, Y., Altinel, M., Franklin, M. J., Zhang, H., and Fischer, P. 2003. Path sharing and predicate evaluation for high-performance XML filtering. ACM Trans. Database Syst. 28, 4 (Dec. 2003): 467-516.

[17]. Guoli li, S Hou, H A Jacobsen. 2008. Routing of XML and XPath queries in data dissemination networks, ICDCS 2008. Beijing, China: 627-638.

[18]. Petrovic, M., Liu, H., and Jacobsen, H. 2005. G-ToPSS: fast filtering of graph-based metadata. WWW 2005. Chiba, Japan: 539-547.

[19]. E. Fidler, H. A. Jacobsen, G. Li, and S. Mankovski. 2005. The PADRES Distributed Publish/Subscribe System. FIW 2005. Leicester, UK: 12–30.

[20]. P. R. Pietzuch, B. Shand, and J. Bacon. 2004. Composite event detection as a generic middleware extension. IEEE Network Magazine, Special Issue on Middleware Technologies for Future Communication Networks, January/February 2004: 44-55.

[21]. Guoli Li and Hans-Arno Jacobson. Composite subscriptions in content based publish/subscribe systems. 2005. In Proceedings of the Sixth International Conference on Middleware. Grenoble, France: 249-269

[22]. R. Baldoni, C. Marchetti, A. Virgillito, and R. Vitenberg. 2005. Content-Based Publish-Subscribe over Structured Overlay Networks. ICDCS 2005. Columbus, Ohio, USA: 437–446.

[23]. A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. 2001. Design and evaluation of a wide-area event notification service. ACM ToCS. 19 (3): 332–383.

[24]. Guoli Li, Vinod Muthusamy, and Hans-Arno Jacobsen. 2008. Adaptive Content-Based Routing in General Overlay Topologies, Middleware 2008. Leuven, Belgium: 1-21.

[25]. S. Tarkoma and J. Kangasharju. 2005. Filter Merging for Efficient Information Dissemination. Springer Volume 3760 of Lecture Notes in Computer Science: 274–291.

[26]. A. Riabov, Z. Liu, J. L. Wolf, P. S. Yu, and L. Zhang. 2002. Clustering algorithms for content-based publication-subscription systems. ICDCS 2002. Vienna, Austria: 133-142

[27]. Y.-M. Wang, L. Qiu, D. Achlioptas, G. Das, P. Larson, and H. J.Wang. 2002. Subscription partitioning and routing in content-based publish/subscribe systems. In Proceedings of the 16th International Symposium on Distributed Computing (DISC), Toulouse, France

[28]. Eugene Wu, Yanlei Diao, and Shariq Rizvi. 2006. High-performance complex event processing over streams. In SIGMOD2006. Chicago, Illinois, USA:407-418

[29]. Jagrati Agrawal, Yanlei Diao, Daniel Gyllstrom, and Neil Immerman. 2008. Efficient pattern matching over event streams. In SIGMOD 2008. Vancouver, BC, Canada: 147-160

[30]. Fusheng Wang, Shaorong Liu, Peiya Liu and Yijian Bai. 2006. Bridging physical and virtual worlds: complex event processing for RFID data streams. Lecture Notes in Computer Science, Volume 3896/2006: 588-607.

[31]. Nicholas Poul Schultz-Moller, Matteo Migliavacca, and Peter Pietzuch. 2009. Distributed complex event processing with query rewriting. In DEBS 2009. Nashville, Tennessee, USA. 4:1-4:12.

[32]. BEA Suvey, 2007. Event Processing Market Pulse 2007, On the 3rd EPTS (Event Processing Technology Society) 17-19 Sept. 2007

[33]. http://www.research.ibm.com/distributedmessaging/gryphon.html

[34]. http://padres.msrg.utoronto.ca

作者简介：

虎嵩林： 中国科学院计算技术研究所前瞻研究实验室副研究员 husonglin@ict.ac.cn